AMENDMENTS TO THE CLAIMS

Please cancel claim 19 without prejudice or waiver to the underlying subject matter.

1. **(Original)** A computer system for generating output modules in a form-based application runtime environment, comprising:

    a form manager component configured to receive an indication that a reusable form element has been changed, determine which output modules from a set of output modules are affected by the changed form element, and invalidate the affected output modules; and

    a runtime manager component configured to receive a request for an output module from the set of output modules and cause regeneration of the requested output module if the requested output module has been invalidated.

2. **(Original)** The system of claim 1, wherein the indication is received when changes to the reusable form element are saved.

3. **(Original)** The system of claim 1, wherein the affected output modules are determined by referencing a record data structure.

4. **(Original)** The system of claim 1, wherein the affected output modules are invalidated by marking a flag associated with each affected output module as invalid.

5. **(Original)** The system of claim 1, wherein the request for the output module received by the runtime manager is a request to identify the output module.

6. **(Original)** The system of claim 1, wherein the reusable form element is one of a form page and a form window.

7. **(Original)** The system of claim 1, wherein the reusable form element is form logic.

8. **(Original)** The system of claim 1, wherein the reusable form element is a form interface.

9. **(Original)** A computer-implemented method for generating output modules in a form-based application runtime environment, comprising:

receiving an indication that a reusable form element has been changed;

determining which output modules from a set of output modules are affected by the changed form element;

invalidating the affected output modules;

receiving a request for an output module from the set of output modules; and

regenerating the requested output module if the requested output module has been invalidated.

10. **(Original)** The method of claim 9, wherein the indication is received when changes to the reusable form element are saved.

11. **(Original)** The method of claim 9, wherein the affected output modules are determined by referencing a record data structure.

12. **(Original)** The method of claim 9, wherein the affected output modules are invalidated by marking a flag associated with each affected output module as invalid.

13. **(Original)** The method of claim 9, wherein the request for the output module received by the runtime manager is a request to identify the output module.

14. **(Original)** The method of claim 9, wherein the reusable form element is one of a form page and a form window.

15. **(Original)** The method of claim 9, wherein the reusable form element is form logic.

16. **(Original)** The method of claim 9, wherein the reusable form element is a form interface.

17. **(Original)** A computer-implemented dynamic form building method, comprising:

responsive to a call to start a form output process based on an identified form:

determining whether a previously generated output module associated with the identified form in an output module library has been marked as invalid;

if so:

regenerating the output module; and

storing the regenerated output module in the output module library along with a marker to indicate that the output module is valid.

18. **(Original)** The method of claim 17, wherein the regeneration of the output module includes compiling changed reusable form elements into the output module.

19. **(Canceled)**